

Amendments to the Claims:

The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claims 1-22 (Previously Canceled)

Claim 23 (Canceled)

Claims 24-25 (Previously Canceled)

26. (Previously Presented) A method of processing one or more custom action tags in a Web Page to provide a tag library extension that can be used to implement a tag library supporting multiple scripting languages, said custom action tag providing textual instructions for performing actions in said Web Page, said custom action tag being capable of customization to perform customized actions and including a start-tag and an end-tag, said custom action tag being further capable of having a body between said start-tag and said end-tag, said method comprising:

providing, for said custom action tag, a tag-handler object that represents a run-time representation of said custom action tag, said tag-handler object including:

a do-start method for processing said start-tag of said custom action tag,

a do-body method for processing said body of said custom action tag, and

a do-end method for processing of said end-tag of said custom action tag;

invoking said do-start method of said tag-handler object to process said start-tag of said custom action tag, wherein said do-start method determines:

whether the custom action tag has a body, and

whether there is a need to process said body when said do-start method determines that said custom action tag has a body;

invoking said do-body method of said tag-handler object when said invoking of said do-start method determines that there is a need to process said body of said custom action tag;

processing, by said do-body method of said tag-handler object, said body of said custom action tag, to translate said body from a first scripting language to platform independent code that can be executed to perform actions intended by said custom action tag;

determining by said do-body method of said tag-handler object whether further processing is required to translate the body from a first scripting language to platform independent code that can be executed to perform the actions intended by said custom action tag when said processing has been performed by said do-body method of said tag-handler object,

repeating said processing, by said do-body method of said tag-handler object, when said do-body method of said tag-handler object determines that further processing is required; and

invoking said do-end method of said tag-handler object when said do-body method determines that no further processing is required, wherein said do-end method processes said end-tag of said custom action tag.

27. (Previously Presented) A method as recited in claim 26, wherein said method further comprises:

creating at runtime when said platform independent code is to be executed, by said tag-handler, one or more objects that said Web Page requires;

storing at runtime when said platform independent code is to be executed, by said tag-handler at runtime, said one or more created objects into the pageContext object, thereby allowing the one or more objects to be retrieved at runtime.

28. (Previously Presented) A method as recited in claim 26,

wherein said invoking of said do-start method of said tag-handler object provides said do-start method with one or more attribute values associated with said custom action tag, and

wherein said method further comprises:

providing a page context object that provides runtime data including runtime values for one or more attributes associated with said custom action tag; and

interacting, by said do-start method, with said page context object to perform appropriate functionality with respect to said attribute values.

29. (Previously Presented) A method as recited in claim 26, wherein said processing of said custom action tag by said do-body method comprises:

evaluating said body of said custom action tag as a stream of bytes that can be represented as a body-evaluation object, said body-evaluation object providing an abstraction from various scripting languages that can be used to implement said custom action tag.

30. (Previously Presented) A method as recited in claim 26, wherein the body-evaluation object is implemented using a buffer.

31. (Previously Presented) A method as recited in claim 29, wherein said evaluating of said custom action tag by said do-body method further comprises:

providing said do-body method with said body-evaluation object;

determining by said do-body method whether to insert said stream of bytes into another stream of bytes;

inserting, by said do-body method, said stream of bytes into another stream of bytes, when said do-body method determines that said stream of bytes should be inserted into another stream of bytes; and

converting, by said do-body method, said stream of bytes to a string when said do-body method determines that said stream of bytes should not be inserted into another stream of bytes.

32. (Previously Presented) A method as recited in claim 31, wherein said do-end method:

facilitates cleanup and update of said context page; and

determines whether further processing is required for another custom action tag.

33. (Previously Presented) A method as recited in claim 26, wherein the do-body method of said tag-handler object comprises:

a do-init-body method that operates to initialize and pre-process a body-evaluation object that provides an abstraction from a scripting language used to implement said custom action tag; and

a do-after-body method that operates to evaluate said body-evaluation object at least once to translate the body from a first scripting language to platform independent code that can be executed to perform the actions intended by said custom action tag.

34. (Previously Presented) A method as recited in claim 26, wherein said Web Page is implemented on a server that supports a platform independent programming language.

35. (Previously Presented) A computer readable media including computer program code for processing one or more custom action tags in a Web Page to provide a tag library extension that can be used to implement a tag library supporting multiple scripting languages, said custom action tag providing textual instructions for performing actions in said Web Page, said custom action tag being capable of customization to perform customized actions and including a start-tag and an end-tag, said custom action tag being further capable of having a body between said start-tag and said end-tag, said readable media comprising;

computer program code for providing, for said custom action tag, a tag-handler object that represents a run-time representation of said custom action tag, said tag-handler object including:

a do-start method for processing said start-tag of said custom action tag,
a do-body method for processing said body of said custom action tag, and
a do-end method for processing of said end-tag of said custom action tag;
computer program code for invoking said do-start method of said tag-handler object to process said start-tag of said custom action tag, wherein said do-start method determines:

whether the custom action tag has a body, and

whether there is a need to process said body when said do-start method determines that said custom action tag has a body;

computer program code for invoking said do-body method of said tag-handler object when said invoking of said do-start method determines that there is a need to process said body of said custom action tag;

computer program code for processing, by said do-body method of said tag-handler object, said body of said custom action tag, to translate said body from a first scripting language to platform independent code that can be executed to perform actions intended by said custom action tag;

computer program code for determining by said do-body method of said tag-handler object whether further processing is required to translate the body from a first scripting language to platform independent code that can be executed to perform the

actions intended by said custom action tag when said processing has been performed by said do-body method of said tag-handler object,

computer program code for repeating said processing, by said do-body method of said tag-handler object, when said do-body method of said tag-handler object determines that further processing is required; and

computer program code for invoking said do-end method of said tag-handler object when said do-body method determines that no further processing is required, wherein said do-end method processes said end-tag of said custom action tag.

36. (Previously Presented) A computer readable medium as recited in claim 35,

wherein said invoking of said do-start method of said tag-handler object provides said do-start method with one or more attribute values associated with said custom action tag, and

wherein said computer readable medium further comprises:

computer program code for providing a page context object that provides runtime data including run-time values for one or more attributes associated with said custom action tag; and

computer program code for interacting, by said do-start method, with said page context object to perform appropriate functionality with respect to said attribute values.

37. (Previously Presented) A computer readable medium as recited in claim 35, wherein said processing of said custom action tag by said do-body method comprises:

evaluating said body of said custom action tag as a stream of bytes that can be represented as a body-evaluation object, said body-evaluation object providing an abstraction from various scripting languages that can be used to implement said custom action tag.

38. (Previously Presented) A computer readable medium as recited in claim 37, wherein the body-evaluation object is implemented using a buffer.

39. (Previously Presented) A computer readable medium as recited in claim 38, wherein said evaluating of said custom action tag by said do-body method further comprises:

providing said do-body method with said body-evaluation object;

determining by said do-body method whether to insert said stream of bytes into another stream of bytes;

inserting, by said do-body method, said stream of bytes into another stream of bytes, when said do-body method determines that said stream of bytes should be inserted into another stream of bytes; and

converting, by said do-body method, said stream of bytes to a string when said do-body method determines that said stream of bytes should not be inserted into another stream of bytes.

40. (Previously Presented) A computer system for processing one or more custom action tags in a Web Page to provide a tag library extension that can be used to implement a tag library supporting multiple scripting languages, said custom action tag providing textual instructions for performing actions in said Web Page, said custom action tag being capable of customization to perform customized actions and including a start-tag and an end-tag, said custom action tag being further capable of having a body between said start-tag and said end-tag, said computer system comprising;

at least one central processing unit;

memory; and

a computer program operating on said at least one central processing unit, wherein said computer program is capable of:

providing, for said custom action tag, a tag-handler object that represents a run-time representation of said custom action tag, said tag-handler object including:

a do-start method for processing said start-tag of said custom action tag,

a do-body method for processing said body of said custom action tag, and

a do-end method for processing of said end-tag of said custom action tag;

invoking said do-start method of said tag-handler object to process said start-tag of said custom action tag, wherein said do-start method determines:

whether the custom action tag has a body; and

whether there is a need to process said body when said do-start method determines that said custom action tag has a body;

invoking said do-body method of said tag-handler object when said invoking of said do-start method determines that there is a need to process said body of said custom action tag;

processing, by said do-body method of said tag-handler object, said body of said custom action tag, to translate said body from a first scripting language to platform independent code that can be executed to perform actions intended by said custom action tag;

determining by said do-body method of said tag-handler object whether further processing is required to translate the body from a first scripting language to platform independent code that can be executed to perform the actions intended by said custom action tag when said processing has been performed by said do-body method of said tag-handler object;

repeating said processing, by said do-body method of said tag-handler object, when said do-body method of said tag-handler object determines that further processing is required; and

invoking said do-end method of said tag-handler object when said do-body method determines that no further processing is required, wherein said do-end method processes said end-tag of said custom action tag.

41. (Previously Presented) A computer system as recited in claim 40, wherein said Web Page is implemented on a server that supports a platform independent programming language.

42. (New) A method for translating an action tag in a JavaServer Page, wherein the method provides support for multiple scripting languages, the method comprising:

- (a) generating and storing code for any attribute values in a tag instance;
- (b) instantiating the tag;
- (c) invoking a doStart () method;
- (d) creating a BodyEvaluation object if necessary;
- (e) If a (evaluate_body) result is true, then
 - (i) invoking a doBody () method on the BodyEvaluation object;
 - (ii) if the doBody() method returns true, then goto (e)
- (f) invoking a doEnd () method on the tag of the tag instance; and
- (g) synchronizing any indicated scripting variables.